

CLAIMS

WHAT IS CLAIMED IS:

1. A computer-readable medium having encoded thereon a data structure which represents hierarchically-organized data, said hierarchically-organized data having at least a first node at a first level and a plurality of second nodes at a second level, the second nodes being child nodes of the first node, the first and second nodes each having a corresponding data item associated therewith, the data structure comprising:

a plurality of rows each having a plurality of fields, each of said rows corresponding to a data item associated with a one of the first and second nodes, the fields of each row comprising:

a first field which stores the data item associated with the one of the nodes that corresponds to the row; and

a second field which stores a position identifier which identifies the level at which the node that corresponds to the row is located in the hierarchically-organized data, and which further indicates one of: (a) an identity of an ancestor node of the node that corresponds to the row, or (b) the fact that the note that corresponds to the row has no ancestor.

2. The computer-readable medium of claim 1, wherein the data structure comprises a relation in a relational database.

3. The computer-readable medium of claim 1, wherein the hierarchically-organized data comprises data in a hierarchical markup language.

4. The computer-readable medium of claim 3, wherein said hierarchical markup language comprises eXtensible Markup Language (XML).

5. The computer-readable medium of claim 4, wherein the fields of each row further comprise:

a name identifier identifying a user-assigned XML name; and
a data type.

5

6. The computer-readable medium of claim 1, wherein the position identifier of the first node comprises a first value in a space of ordered values, and wherein the position identifiers of each of the second nodes comprises said first value and a second value in said space of ordered values.

10

7. The computer-readable medium of claim 6, wherein an order is defined among the second nodes, and wherein the second values associated with the second nodes are respective of said order with respect to said space of ordered values.

15

8. The computer-readable medium of claim 7, wherein said space of ordered values comprises the set of integers, wherein said first value is a "1", and wherein the second values for the second nodes are integers in an increasing series of integers.

20

9. The computer-readable medium of claim 7, wherein said space of ordered values comprises the set of integers, wherein said first value is a "1", and wherein the second values for the second nodes are odd integers in an increasing series of integers.

10. The computer-readable medium of claim 9, wherein said hierarchically-organized data comprises a third node which is a child of said first node and which is located between first and second ones of said second nodes, said third node having a position identifier associated therewith, and wherein the position identifier for the second node comprises said first value, an even number between the second values associated with said first and second ones of said second nodes, and an odd number.

25

11. The computer-readable medium of claim 6, wherein said ordered values are represented in a form comprising:

- a length indicator selected from a length-indicator space having non-uniform numbers of bits, each length indicator in said length-indicator space having a prefix property such that no member of said length-indicator space is a prefix of any other member of said length-indicator space; and
- an ordinal indicator having a length indicated by said length indicator.

12. The computer-readable medium of claim 11, wherein each of the position identifiers comprises a bit length field indicative of the aggregate number of bits for the length indicators and the position identifiers in the position identifier of the corresponding bit length field.

13. A method of representing hierarchically-organized data, the hierarchically-organized data comprising at least a first node at a first level and a plurality of second nodes at a second level, the second nodes being child nodes of the first node, an order being defined among the second nodes, the first and second nodes each having a data item associated therewith, the method comprising:

- assigning a first position identifier to the first node, wherein said first position identifier comprises a first value selected from an ordered space of values;
- assigning a second position identifier to each of the second nodes, each of the second identifiers comprising said first value and a second value selected from said ordered space of values, wherein the second values are assigned to the second nodes respectively of the order; and
- storing, in a non-hierarchical data structure, a plurality of data records, wherein each of the plurality of data records corresponds to one of the first or second nodes, and wherein each data record includes, for its corresponding node:

the position identifier associated with the node; and
the data item associated with the node.

14. The method of claim 13, wherein said non-hierarchical data structure
5 comprises a relational database, and wherein each of said data records comprises a row
of a relation in said relational database.

15. The method of claim 13, wherein said hierarchically-organized data
comprises data in eXtensible Markup Language (XML) having a plurality of tags, each
10 of said tags delimiting a portion of the hierarchically-organized data, said tags being
nestable, and each of said tags and its delimited data corresponding to one or more
nodes in the hierarchically-organized data.

16. A system for storing hierarchically-organized data, the hierarchically-
15 organized data comprising at least a first node at a first level and a plurality of second
nodes at a second level, the second nodes being child nodes of the first node, each of
the first and second nodes having a data item associated therewith, the system
comprising:

a relational table having a plurality of rows, each of said rows corresponding
20 to a node in the hierarchically-organized data, the relational table having a plurality of
columns, the columns comprising:

a first column which stores a position identifier indicative of the level at
which the node that corresponds to the row is located in the hierarchically-organized
data, and further indicative of an ancestor of the node that corresponds to the row; and

25 a second column which stores the data item associated with the node that
corresponds to the row; and

a relational table manager which inserts and retrieves the rows from the
relational database.

17. The system of claim 16, wherein said relational table manager comprises a database management system.

5 18. The system of claim 16, wherein the hierarchically-organized data comprises eXtensible Markup Language (XML) data.

19. The system of claim 16, wherein said XML data comprises a plurality of tags which delimit portions of the hierarchically-organized data, said tags being
10 nestable, wherein each of said tags and its corresponding delimited data corresponds to one or more nodes in the hierarchically-organized data, and at least some of the rows in said relational table further comprise:

 a tag identifier indicative of the tag associated with the node that corresponds to the row; and

15 a type identifier indicative of a type of the data delimited by the tag indicated by the row's tag identifier.

20 20. A method of inserting a new node into a hierarchically-organized data structure, the hierarchically-organized data structure comprising at least a first node at a first level and a plurality of second nodes at a second level, the second nodes being child nodes of the first node, an order being defined among the second nodes, each of the nodes having a data item associated therewith, the first node having a first position identifier which comprises a value selected from a space of ordered discrete values, the second nodes having second position identifiers, each of the second position identifiers
25 comprising the first value and a second value selected from a first subset of the space of ordered discrete values, the first subset consisting of non-adjacent values in the space of ordered discrete values, the second values assigned to each of the second nodes being respective of the order, the method comprising the steps of:

receiving the new node and an indication of a position in the hierarchically-organized data structure into which the new node is to be inserted, said position being located between first and second ones of the second nodes with respect to the order; and assigning the new node a third position identifier which comprises:

5 said first value;

 a third value selected from a second subset of the space of ordered discrete values, the second subset consisting of the difference between the space of ordered discrete values and the first subset, said third value being located between the second values associated with said first and second ones of said second nodes with
10 respect to the order; and

 a fourth value selected from the first subset.

21. The method of claim 20, further comprising:

 storing in a relation a row which comprises:

15 said third position identifier; and

 a data item associated with the new node.

22. The method of claim 21, wherein said relation further stores a row corresponding to each of the first and second nodes, each of the rows comprising the
20 position identifier of the corresponding first or second node, and the data item of the corresponding first or second node.

23. The method of claim 20, wherein said hierarchically organized data comprises eXtensible Markup Language (XML) data comprising a plurality of tags
25 which delimit portions of the hierarchically-organized data, said tags being nestable, wherein each of said tags and its corresponding delimited data corresponding to one or more nodes in the hierarchically-organized data.

24. A computer-readable medium having computer-executable instructions, the instructions operating on hierarchically-organized data, the hierarchically-organized data having at least a first node at a first level and a plurality of second nodes at a second level, the second nodes being child nodes of the first node, each of the first and
5 second nodes having a data item associated therewith, the instructions being adapted to perform acts comprising:

traversing the hierarchically-organized data in pre-order;

assigning an identifier to each of the first and second nodes in a sequence corresponding to an order in which the first and second nodes are encountered during
10 said traversing act;

storing a plurality of records in a non-hierarchical data structure, each of said records corresponding to one of the first or second nodes, each of said records comprising:

the identifier assigned to the node that corresponds to the row; and
15 the data item associated with the node that corresponds to the row;
receiving a new node to be inserted into the hierarchically-organized data;
assigning, to the new node, a new identifier which comprises the identifier assigned to a one of the first and second nodes and a value; and
storing, in the non-hierarchical data structure, a new record corresponding
20 said new node, said new record comprising said new identifier and a data item associated with said new node.

25. The computer-readable medium of claim 24, further comprising:

receiving an indication that said new node is to be either a child or a sibling
25 of said one of said second nodes.

26. The computer-readable medium of claim 24, wherein said assigning act comprises assigning successive integers to each of the nodes encountered in said traversing act.

5 27. The computer-readable medium of claim 24, wherein said hierarchically-organized data comprises a tree data structure.

28. The computer-readable medium of claim 27, wherein said tree data structure comprises eXtensible Markup Language (XML) data having a plurality of portions
10 delimited by a plurality of tags, each of the tags and its delimited data corresponding to one or more nodes in the hierarchically-organized data structure.

29. A method of comparing the relative position of a first and second nodes in a hierarchically-organized data structure, the hierarchically-organized data structure
15 comprising at least a first node at a first level, a plurality of second nodes at a second level, and a plurality of third nodes at a third level, a first group of the third nodes being child nodes of a first one of the second nodes, and a second group of the third nodes being child nodes of a second one of the second nodes, an order being defined among the first, second, and third nodes, the method comprising:

20 assigning a first identifier to the first node, said first identifier comprising first value selected from a space of ordered values;

 assigning a plurality of second identifiers to the second nodes, the identifier for each of the second nodes comprising said first value and a second value selected from said space of ordered values, said second values being assigned to the second
25 nodes respective of the order;

 assigning a plurality of third identifiers to the first group of the third nodes, the third identifier for each of the first group of third nodes comprising said first value, the second value associated with the first one of the second nodes, and a third value

selected from said space of ordered values, said third values being assigned to the first group of third nodes respective of the order;

assigning a plurality of third identifiers to the second group of the third nodes, the third identifier for each of the second group of third nodes comprising said first value, the second value associated with the second one of the second nodes, and a third value selected from said space of ordered values, said third values being assigned to the first group of third nodes respective of the order;

identifying a fourth and fifth node from among the first, second, and third nodes;

comparing corresponding portions of the identifiers of said fourth and fifth nodes until a pair of corresponding non-equal identifiers is encountered;

determining that the encountered portion in the fourth node appears earlier in the space of ordered values than the corresponding encountered portion in the fifth node; and

determining that the fourth node appears earlier in the order than the fifth node.

30. The method of claim 29, wherein said space of ordered values comprises the integers.

31. The method of claim 29, wherein the identifiers associated with each of the nodes comprises one or more variable-length bit sequences, and wherein said comparing act comprises performing a bitwise comparison of the identifiers associated with the fourth and fifth nodes.

32. The method of claim 29, wherein each of the portions comprises a value in said space of ordered values.